

능동 카메라를 이용한 이동 물체 추적

박현석[○] 한종원 조진수 이일병
연세대학교 컴퓨터 과학과

{blackstar[○], zeross, hamster, yblee}@csai.yonsei.ac.kr

Moving object tracking using active camera

Hyunsuk Park[○], Jongwon Han, Jinsu Jo, Yillbyung Lee
Dept. of Computer Science, Yonsei University

요 약

본 논문에서는 인간의 눈 움직임이 반영된 물체 추적 기능을 모방하여 CCD 카메라를 통하여 실시간으로 입력되는 영상 데이터로부터 특징기반 정합방법을 응용하여 움직임 정보를 추출한 후, 팬-틸트(pan-tilt) 기능의 하드웨어를 제어하여 실시간으로 이동하는 물체를 효율적으로 추적하는 시스템을 제안하였다. 기존의 연구들에서는 주로 물체의 색상값을 이용하여 추적이 이루어지므로 조명이나 카메라의 변화에 따라 이동 물체를 놓치거나 유사한 색의 다른 물체를 잘못 추적하는 문제가 있었다. 이러한 문제점을 해결하기 위하여 특징기반의 정합을 응용하여 이동하는 카메라에서 이동물체를 추출하고 이 이동 물체의 좌표를 이용하여 팬-틸트 하드웨어를 제어하여 추적을 수행하였다. 실험 결과 본 시스템은 움직이는 물체를 감지해서 팬-틸트 하드웨어를 올바르게 제어하며 카메라의 움직임을 보정해가며 전체적으로 움직이는 영상 내에서 실제 움직이는 물체를 일관성 있게 추적하는 만족스러운 결과를 보인다.

1. 서 론

특수 환경에서 이루어지는 작업의 원활한 수행을 위해서는 사람을 대신하여 작업을 수행할 수 있는 로봇을 이용한 원격 및 자동화 작업이 필수적이다. 이러한 로봇 비전은 인간의 인지정보의 70%이상을 담당하는 시각정보를 처리하여 로봇에게 시각기능을 부여하기 위한 기술이다. 인간은 시각정보를 통해 여러 가지 상황을 인지하고 판단하고 학습한다. 그중에 가장 중요한 것은 3차원 공간에서 입체감을 표현하는 것과 목표물까지의 거리를 인지하는 능력, 즉, 공간인지능력이라 할 수 있다. 인간의 시각계에서 목표물까지의 거리를 인지함으로써 차후 행동에 대한 의사 결정을 할 수 있으므로 인간의 시각 시스템을 컴퓨터에 적용할 때 중요한 것이 목표물까지의 거리 측정임을 알 수 있다. 이러한 비전을 이용한 거리 측정에서 다른 장비들을 사용하지 않고 값싼 CCD 카메라만을 이용하여 정확한 거리를 측정할 수 있다면 앞으로 로봇의 대중화에도 크게 기여할 것이다.

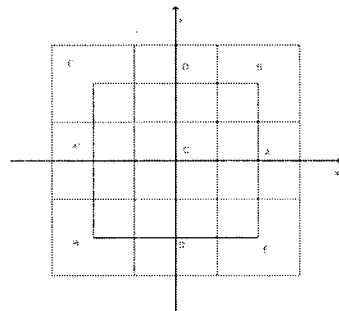
2. 특징점 추출

특징기반 매칭기술은 영상에서의 에지(edge), 정점(vertex),윤곽(contour)등과 같이 주변 화소에 비해 뚜렷이 구분되는 점으로 특징을 추출 후 추출된 특징점에 대응하는 탐색영상의 특징점을 구한다.

모서리 검출기법(Corner detection)은 추적(tracking), 영상 매칭(image matching), 영상인식(image recognition)

등 영상 처리에 많이 사용되어 지며 영상처리의 첫 번째 과정이기도 하다. 현재 많은 모서리 검출기법 알고리즘들이 나와 있는데 대표적인 Harris 모서리검출 알고리즘과 Trajkovic의 모서리검출 알고리즘을 비교하여 보았다.[5][6]

Harris 모서리검출 방법은 널리 쓰이는 방법이기도 하지만 잡음에 민감하고 계산량이 많다는 단점이 있다. 노이즈를 줄이기 위해서는 더 큰 윈도우를 사용하면 되나 그에 따른 계산량이 더욱 많아지고 edge와 같은 중요한 정보가 손실되는 문제가 생긴다. 따라서 실시간 처리에 적합하지 않다. Trajkovic operator를 사용하면 계산량이 적고 시간이 빨라 실시간에 적합하나 회전에 약하다는 단점이 있다. 본 논문에서는 Trajkovic 8-neighbors 방법을 이용하여 특징점을 추출하였다.



[그림1]Trajkovic 8-neighbor operator 3*3 윈도우

$$C_{SAMPLE}(x, y) = \min(r_A, r_B, r_D, r_E) \dots\dots\dots \text{식 1}$$

where:

$$\begin{aligned} r_A &= (I_A - I_D)^2 + (I_A - I_C)^2 \\ r_B &= (I_B - I_D)^2 + (I_B - I_C)^2 \\ r_D &= (I_D - I_C)^2 + (I_D - I_C)^2 \\ r_E &= (I_E - I_C)^2 + (I_E - I_C)^2 \end{aligned}$$

여기서 I_A, I_B, I_D, I_E 는 각각 [그림 1]의 A,B,D,E와 대응한다.

식 1 과 같은 방법으로 각 r_A, r_B, r_D, r_E 를 계산한 후 최소값을 C_{SAMPLE} 로 선택하고 특징점의 후보를 찾고 $C_{SAMPLE} \geq (\text{Threshold})$ 이상이면 특징점 후보로 등록한다.

$$C_{INTERPIXEL}(x, y) = \min(r_1(x), r_2(x), r_3(x), r_4(x))$$

where:

$$\begin{aligned} r_1(x) &= A_1x^2 + 2B_1x + r_A \\ r_2(x) &= A_2x^2 + 2B_2x + r_B \\ r_3(x) &= A_3x^2 + 2B_3x + r_D \\ r_4(x) &= A_4x^2 + 2B_4x + r_E \\ B_1 &= (I_B - I_A)(I_A - I_C) + (I_B - I_A)(I_A - I_C) \\ B_2 &= (I_D - I_B)(I_B - I_C) + (I_D - I_E)(I_B - I_C) \\ B_3 &= (I_E - I_D)(I_E - I_C) + (I_E - I_D)(I_E - I_C) \\ B_4 &= (I_A - I_E)(I_E - I_C) + (I_A - I_E)(I_A - I_C) \\ A_1 &= r_B - r_A - 2B_1 \\ A_2 &= r_D - r_B - 2B_2 \\ A_3 &= r_E - r_D - 2B_3 \\ A_4 &= r_A - r_E - 2B_4 \end{aligned}$$

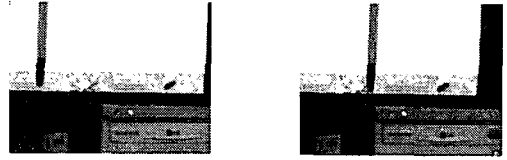
$$C_{INTERPIXEL}(x, y) =$$

$$\begin{cases} C_{SAMPLE}(x, y) & \text{if for all } i=1,2,3,4 \\ & \text{either } B_i \geq 0 \text{ or } A_i + B_i \leq 0 \\ \min \left[r_i - \frac{B_i^2}{A_i} \right] & \text{for all } i=1,2,3,4, \\ & \text{that satisfy } B_i < 0 \text{ and } A_i + B_i > 0 \end{cases}$$

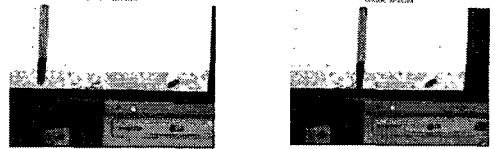
.....식 2

where : $r_1 = r_A, r_2 = r_B, r_3 = r_D, r_4 = r_E$

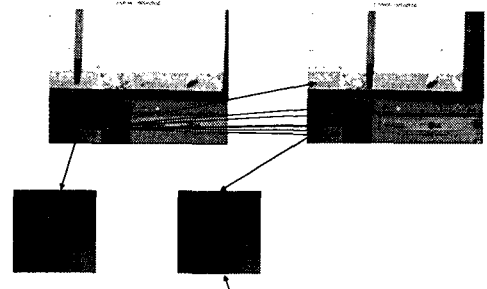
식2와 같은 방법으로 특징점 추출을 한 후 지역 최대값을 찾는다. t_0 frame(이전 이미지)과 $t_0 + \Delta t$ frame 을 Trajkovic 방법을 이용하여 특징점을 추출한 후 추출된 좌표를 이용하여 correlation matching을 수행한다.



(a) (b)
[그림2] 입력영상 (a)Left t_0 image (b)Left $t_0 + \Delta t$ image

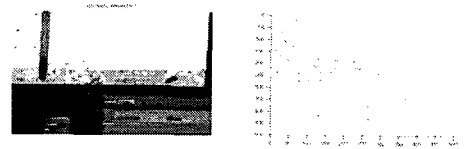


(a) (b)
[그림3] 각각 입력영상(a), (b)에서 특징점 추출

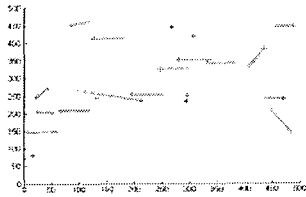


[그림4] 11*11상관관계정합(Correlation matching)

[그림3]에서 특징점 추출을 한 결과 (a)이미지에서 48개의 특징점이 나타나고 (b)에서 50개의 특징점이 나타났다. [그림4]는 11*11크기의 윈도우로 상관관계 정합을 하는 그림이다. [그림5]는 11*11의 윈도우 크기로 상관관계 정합을 한 후 매칭쌍을 선으로 연결시켜놓은 그림이다. 그림 (b)에서와 같이 1:1매칭으로 나타나지 않고 여러 개의 매칭쌍이 나타났다. 여기서 1:1매칭이 되지 않는 매칭쌍을 후보에서 제외한다. [그림6]은 1:1매칭이 아닌 정합쌍을 제외한 결과이고 +로 나타난 점은 후보에서 제외된 점이다.



(a) (b)
[그림5](a) 상관관계 정합 후 정합쌍을 선으로 연결한 영상 (b)정합쌍을 선으로 연결



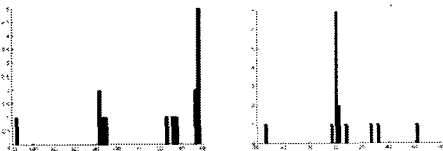
[그림6] 1:1매칭이 아닌 정합쌍을 제외한 결과

3. 이동거리 계산

각 매칭쌍의 이동거리를 x축과 y축으로 나누어 히스토그램으로 나타내면 [그림7]과 같이 나타난다.

[표 1] t_0 영상과 $t_0 + \Delta t$ 영상의 정합쌍의 거리 값 추출

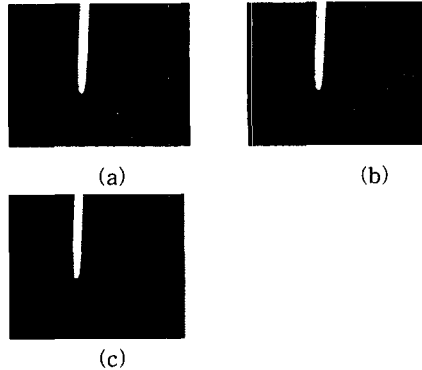
| | x축거리 | y축거리 |
|----|------|------|
| 1 | 56 | 1 |
| 2 | 32 | -2 |
| 3 | 54 | 0 |
| 4 | -37 | 63 |
| 5 | 25 | 27 |
| 6 | -117 | 32 |
| 7 | 57 | 0 |
| 8 | -36 | 2 |
| 9 | 53 | 3 |
| 10 | -30 | -54 |
| 11 | 58 | 0 |
| 12 | 56 | 0 |
| 13 | 56 | 1 |
| 14 | 35 | 9 |
| 15 | -34 | 0 |



[그림7] (a) x축 이동거리 히스토그램 (b) y축 이동거리 히스토그램

이미지를 블록으로 나눈 후 식 3과 같이 블록내의 0이 아닌 값을 카운트하여 합이 임계치 이하이면 잡영으로 판단하여 블록에 0을 넣고 임계치 이상이면 움직임으로 판단하여 블록에 1을 넣어 블록지도를 생성한다.

$$M(x,y) = \begin{cases} 1, & \sum_{j=0}^{M-1} \sum_{i=0}^{N-1} (I_{ij}) \geq T, 1 \\ 0, & \text{otherwise} \end{cases} \quad \text{..... 식 3}$$

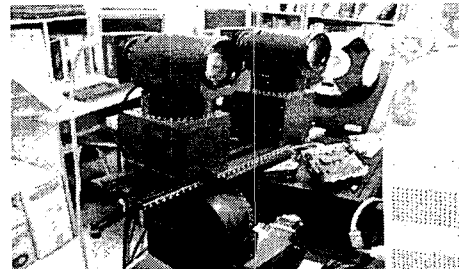


[그림 8] (a)잡영이 나타난 결과영상 (b)블록기반 잡영제거 (c)결과 영상

잡영이 제거된 블록지도를 이용해 컨투어링을 하고 결과 좌표를 얻는다.

4. 결과

본 실험은 능동 카메라에서의 이동물체 검출로 실시간에 동작하도록 설계되었다. 군사용이나 실험결과 초당 7 frame의 처리속도를 보였고 결과 또한 안정적 이었다.



[그림 9] 실험에 사용된 스테레오비전시스템

5.참고문헌

[1] M. Bennamoun and G.J. Mamic, "Object Recognition," springer, pp. 31-33, 2001

[3] A.S. Ogale and Y. Aloimonos, "Shape and the stereo correspondence problem," International Journal of Computer Vision, vol. 65, no. 1, October 2005.

[4] Lipton, Fujiyoshi, Patil, "Moving target classification and tracking from real-time video," IEEE Workshop on Application of Computer vision(WACV), Princeton NJ, pp. 8-14, October. 1998.

[5] Harris, C, and Stephens, M. "A combined corner and edge detector," Proceeding of the Fourth Alvey Vision Conference, pp. 147-151, 1988

[6] Trajkovic, M. and Hedley, M. "Fast corner detection," Image and Vision Computing 16, pp. 75-87 1998